Serial No. 09/846,380
Reply to Office Action of April 4, 2005

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1.   (currently amended) A method for managing a plurality of high-availability-aware applications components in a networked computer system comprising:

        invoking a registration application programming interface by registering the plurality of high-availability-aware applications components to be managed; and

        invoking callback interfaces of registered applications to dynamically allocate allocating roles and assignments to one or more of registered applications components of the plurality of high-availability-aware applications components to achieve a desired redundancy level based on application component type information.

2.   (currently amended) The method of claim 1, further comprising:
        providing information through the application programming interface to the registered applications components so that related applications components among the registered components may communicate to achieve the desired redundancy level.

3.   (currently amended) The method of claim 2, further comprising:
        maintaining software release domain information,
        wherein the software release domain information is provided to the related applications components during the providing step.

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

4.   (currently amended) The method of claim 1, further comprising:

performing administrative actions on the registered <u>applications</u> ~~components~~ in response to a request from an external management agent.

5.   (currently amended) The method of claim 1, further comprising:

responding to an error by changing roles and assignments of the registered ~~components~~ <u>applications via the invocation of the callback interfaces of the registered applications</u>.

6.   (currently amended) The method of claim 5, further comprising:

maintaining <u>application</u> ~~component~~ relationship information,

wherein the <u>application</u> ~~component~~ relationship information is used during the allocating step and the responding step.

7.   (canceled)

8.   (original) The method of claim 5, wherein the responding step uses protection group information.

9.   (currently amended) The method of claim 5, wherein the responding step further comprises:

choosing an appropriate response; and

altering assignments and roles of the registered <u>applications</u> ~~components~~ according to the appropriate response.

10.   (original) The method of claim 9, wherein the appropriate response includes restart, fail-over, switch-over, node fail-over, and node switch-over.

11.   (currently amended) The method of claim 1, wherein the roles allocated to the one or more of the registered <u>applications</u> ~~components~~ include off-line, spare, primary, secondary, and quiescing.

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

12.   (currently amended) The method of claim 1, further comprising:

maintaining <u>application</u> ~~component~~ relationship information,

wherein the <u>application</u> ~~component~~ relationship information is used during

the allocating step.

13.   (canceled)

14.   (original) The method of claim 1, wherein the allocating step uses

protection group information.

15.   (currently amended) The method of claim 1, wherein the allocating step

assigns a specific role and assignment to a self-determining <u>application</u>

~~component~~ in the registered <u>applications</u> ~~components~~.

16.   (currently amended) The method of claim 1, wherein the plurality of high-

availability-aware <u>applications</u> ~~components~~ include stand-alone <u>applications</u>

~~components~~, proxied <u>applications</u> ~~components~~, and proxy <u>applications</u>

~~components~~.

17.   (currently amended) The method of claim 1, wherein the <u>application</u>

~~component~~ type information includes functional attributes, recovery parameter

attributes, <u>application</u> ~~component~~ instance level attributes, and <u>application</u>

~~component~~ assignment level attributes.

18.   (currently amended) A method of allocating an assignment in a

networked computer system comprising;

registering a plurality of components applications <u>through an application</u>

<u>programming interface</u>, wherein the plurality of <u>applications</u> ~~components~~ are

high-availability aware;

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

allocating roles to registered ~~applications~~ ~~components~~ of the plurality of ~~components~~ applications by invoking a callback interface of registered applications;

allocating the assignment to a first application ~~component~~ selected from the registered applications ~~components~~ based on application ~~component~~ type information of the first ~~component~~ application by invoking a callback interface of the first application;

changing a role of the first application ~~component~~ to primary by invoking a callback interface of the first application;

determining [[a]] an application specific redundancy level based on the application ~~component~~ type information; .

allocating the assignment to a predetermined number of secondary applications ~~components~~ selected from the registered applications ~~components~~ based on application ~~component~~ type information of the secondary applications ~~components~~, wherein the predetermined number is based on the redundancy level of the application by invoking a callback interface of the secondary applications;

changing roles of the predetermined number of secondary applications ~~components~~ to secondary by invoking a callback interface of the secondary applications; and

notifying the first ~~component~~ application by invoking a callback interface of the first application about the predetermined number of secondary applications ~~components~~ and the predetermined number of secondary applications ~~components~~ about the first ~~component~~ application by invoking a callback interface of the second applications.

19.   (currently amended) The method of claim 18, further comprising:
detecting an error affecting the first application ~~component~~;

selecting a new primary <u>application</u> ~~component~~ from the predetermined number of secondary <u>applications</u> ~~components~~; and

changing a role of the new primary <u>application</u> ~~component~~ to primary <u>by invoking a callback interface of the new primary application</u>.

20. (currently amended) The method of claim 19, further comprising:

instructing the first ~~component~~ <u>application, by invoking a callback interface of the first application,</u> to communicate information to the new primary <u>application</u> ~~component~~.

21. (currently amended) The method of claim 18, further comprising:

detecting an error affecting the first <u>application</u> ~~component~~; and

restarting the first <u>application</u> ~~component~~.

22. (original) The method of claim 18, further comprising:

maintaining software release domain information,

wherein the software release domain information is included in the notifying step.

23. (currently amended) The method of claim 18, further comprising:

performing administrative actions on the registered <u>applications</u> ~~components~~ in response to a request from an external management agent.

24. (currently amended) The method of claim 18, further comprising:

maintaining <u>application</u> ~~component~~ relationship information;

wherein the <u>application</u> ~~component~~ relationship information is used in the two assignment allocating steps.

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

25.   (currently amended) A method of allocating an assignment to a plurality
of high-availability-aware applications components in a networked computer
system, the method comprising;

registering the plurality of high-availability-aware components applications
through an application programming interface;

allocating roles to registered applications components of the plurality of
high-availability-aware components applications by invoking a callback interface
of the registered applications;

maintaining application component relationship information;

selecting a first application component from the registered applications
components based on application component type information and the
application component relationship information;

allocating the assignment to the first component application by invoking a
callback interface of the first application;

changing a role of the first application component to primary by invoking a
callback interface of the first application;

determining a redundancy level based on the application component type
information;

selecting a predetermined number of secondary applications components
from the registered applications components based on application component
type information of the secondary applications components and the application
component relationship information, wherein the predetermined number is based
on the determined redundancy level;

changing roles of the predetermined number of secondary applications
components to secondary by invoking a callback interface of the secondary
applications; and

notifying the first application, by invoking a callback interface of the first
application, component about the predetermined number of secondary

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

applications ~~components~~ and the predetermined number of secondary

applications, by invoking a callback interface of the secondary applications,

~~components~~ about the first application ~~component~~.

26. (currently amended) The method of claim 25, further comprising:

detecting an error affecting the first application ~~component~~;

selecting a new primary application ~~component~~ from the predetermined

number of secondary applications ~~components~~ using the application ~~component~~

relationship information; and

changing a role of the new primary application ~~component~~ to primary by

invoking a callback interface of the new primary application.

27. (currently amended) The method of claim 26, further comprising:

instructing the first application ~~component~~ to communicate information to

the new primary ~~component~~ application by invoking a callback interface of the

first application.

28. (original) The method of claim 25, further comprising:

maintaining software release domain information, wherein the software

release domain information is included in the notifying step.

29. (currently amended) The method of claim 25, further comprising:

performing administrating actions on the registered applications

~~components~~ in response to a request from an external management agent.

30. (currently amended) A computer program product for managing a

plurality of high-availability-aware applications ~~components~~ in a networked

computer system, the computer program product comprising:

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

computer readable program code configured to register the plurality of high-availability-aware applications components to be managed by invoking a registration application programming interface;

computer readable program code configured to dynamically allocate roles and assignments to one or more [[of]] registered applications components of the plurality of high-availability-aware applications components to achieve a desired redundancy level based on application component type information by invoking a callback interface of the registered applications; and

a computer readable medium having the computer readable program codes embodied therein.

31. (currently amended) The computer program product of claim 30, further comprising:

computer readable program code configured to provide information to the registered applications components so that related applications components may communicate to achieve the desired redundancy level.

32. (currently amended) The computer program product of claim 30, further comprising:

computer readable program code configured to respond to an error by changing roles and assignments of one or more of the plurality of components applications by invoking a callback interface of the registered applications.

33. (currently amended) A computer readable medium configured to embody computer programming instructions for managing a plurality of high-availability-aware applications components in a networked computer system, the computer programming instructions comprising:

registering the plurality of high-availability-aware applications components to be managed through an application programming interface; and

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

dynamically allocating roles and assignments to registered ~~applications~~ <u>applications</u> ~~components~~ of the plurality of high-availability-aware ~~applications~~ <u>applications</u> ~~components~~ to achieve a desired redundancy level based on <u>application</u> ~~component~~ type information <u>by invoking a callback interface of the registered applications</u>.

34.   (currently amended) A computer program product for allocating an assignment in a networked computer system, the computer program product comprising

computer readable program code configured to <u>provide an application programming interface to</u> register a plurality of high-availability-aware <u>applications</u> components;

computer readable program code configured to allocate roles to registered <u>applications</u> components of the plurality of high-availability-aware ~~components~~ <u>applications by invoking a callback interface of the registered applications</u>;

computer readable program code configured to allocate the assignment to a first <u>application</u> ~~component~~ selected from the registered <u>applications</u> ~~components~~ based on <u>application</u> ~~component~~ type information of the first ~~component~~ <u>application by invoking a callback interface of the first application</u>;

computer readable program code configured to change a role of the first <u>application</u> ~~component~~ to primary <u>by invoking a callback interface of the first application</u>;

computer readable program code configured to determine a redundancy level based on the <u>application</u> ~~component~~ type information;

computer readable program code configured to allocate the assignment to a predetermined number of secondary <u>applications</u> ~~components~~ selected from the registered <u>applications</u> ~~components~~ based on <u>application</u> ~~component~~ type information of the secondary components, wherein the predetermined number is

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

based on the redundancy level <u>by invoking a callback interface of the secondary</u> <u>applications</u>;

computer readable program code configured to change roles of the predetermined number of secondary <u>applications</u> ~~components~~ to secondary <u>by</u> <u>invoking a callback interface of the second applications</u>;

computer readable program code configured to notify the first ~~component~~ <u>application by invoking a callback interface of the first application</u> about the predetermined number of secondary <u>applications</u> ~~components~~ and the predetermined number of secondary <u>applications</u> ~~components~~ about the first ~~component~~ <u>application by invoking a callback interface of the secondary</u> <u>applications</u>; and

a computer readable medium having the computer readable program codes embodied therein.

35.   (currently amended) The computer program product of claim 34, further comprising:

computer readable program code configured to detect an error affecting the first <u>application</u> ~~component~~;

computer readable program code configured to select a new primary <u>application</u> ~~component~~ from the predetermined number of secondary <u>applications</u> ~~components~~; and

computer readable program code configured to change a role of the new primary <u>application</u> ~~component~~ to primary <u>by invoking a callback interface of the</u> <u>new primary application</u>.

36.   (currently amended) A system for managing a plurality of high-availability-aware <u>applications</u> ~~components~~ in a networked computer system, the system comprising:

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

means for registering the plurality of high-availability-aware <u>applications</u> ~~components~~ to be managed <u>through an application programming interface</u>; and

means for dynamically allocating roles and assignments to one or more of registered <u>applications</u> ~~components~~ of the plurality of high-availability-aware <u>applications</u> ~~components~~ to achieve a desired redundancy level based on <u>application</u> ~~component~~ type information by invoking a callback interface of the registered applications.[[;]]

37.   (currently amended) The system of claim 36, further comprising:

means for responding to an error by changing roles and assignments of one or more of the registered ~~components~~ <u>applications by invoking a callback interface</u>.

38.   (currently amended) The system of claim 36, further comprising:

means for providing information <u>through the application programming interface</u> to the registered <u>applications</u> ~~components~~ so that related <u>applications</u> ~~components~~ may communicate to achieve the desired redundancy level.

39.   (currently amended) The system of claim 36, further comprising:

means for performing administrative actions on the registered <u>applications</u> ~~components~~ in response to a request from an external management agent.

40.   (currently amended) A system for allocating an assignment in a networked computer system, the system comprising:

means for registering a plurality of high-availability-aware ~~components~~ <u>applications through an application programming interface</u>;

means for allocating roles to registered <u>applications</u> ~~components~~ of the plurality of high-availability-aware ~~components~~ <u>applications by invoking a callback interface of the registered applications</u>;

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

means for allocating the assignment to a first application ~~component~~ selected from the registered applications ~~components~~ based on application ~~component~~ type information of the first ~~component~~ application by invoking a callback interface of the first application;

means for changing a role of the first application ~~component~~ to primary by invoking a callback interface of the first application;

means for determining a redundancy level based on the application ~~component~~ type information;

means for allocating the assignment to a predetermined number of secondary applications ~~components~~ selected from the registered applications ~~components~~ based on application ~~component~~ type information of the secondary applications ~~components~~, wherein the predetermined number is based on the redundancy level by invoking a callback interface of the secondary applications;

means for changing roles of the predetermined number of secondary applications ~~components~~ to secondary by invoking a callback interface of the secondary applications; and

means for notifying the first application ~~component~~ about the predetermined number of secondary ~~components~~ applications by invoking a callback interface of the first application and the predetermined number of secondary applications components about the first ~~component~~ application by invoking a callback interface of the secondary applications.

41.   (currently amended) A mechanism configured to manage a plurality of high-availability-aware applications ~~components~~ in a networked computer system, the mechanism comprising:

a mechanism configured to register through an application programming interface the plurality of high-availability-aware applications ~~components~~ to be managed; and

13

Serial No. 09/846,380
Reply to Office Action of April 4, 2005

a mechanism configured to dynamically allocate roles and assignments to registered <u>applications</u> ~~components~~ of the plurality of high-availability-aware <u>applications</u> ~~components~~ to achieve a desired redundancy level based on <u>application</u> ~~component~~ type information <u>by invoking a callback interface of the registered applications.</u>

42.   (currently amended) The mechanism of claim 41, further comprising:
a mechanism configured to respond to an error by changing roles and assignments of the registered ~~components~~ <u>applications by invoking a callback interface of the registered applications.</u>

43.   (currently amended) The mechanism of claim 41, further comprising:
a mechanism configured to provide information to the registered <u>applications</u> ~~components~~ so that related <u>applications</u> ~~components~~ among the registered <u>applications</u> ~~components~~ may communicate to achieve the desired redundancy level.

44.   (currently amended) The mechanism of claim 41, further comprising:
a mechanism configured to perform administrative actions on the registered <u>applications</u> ~~components~~ in response to a request from an external management agent.

45.   (currently amended) The mechanism of claim 41, further comprising:
a mechanism configured to maintain additional information relevant to managing the registered <u>applications</u> ~~components~~.

46.   (currently amended) The mechanism of claim 45, wherein the additional information includes information regarding software release domains, <u>application</u> ~~component~~ relationships, and protection groups.